Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2316428 A1 2002/02/18

(21) 2 316 428

(12) DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION

(13) A1

(22) Date de dépôt/Filing Date: 2000/08/18

(41) Mise à la disp. pub./Open to Public Insp.: 2002/02/18

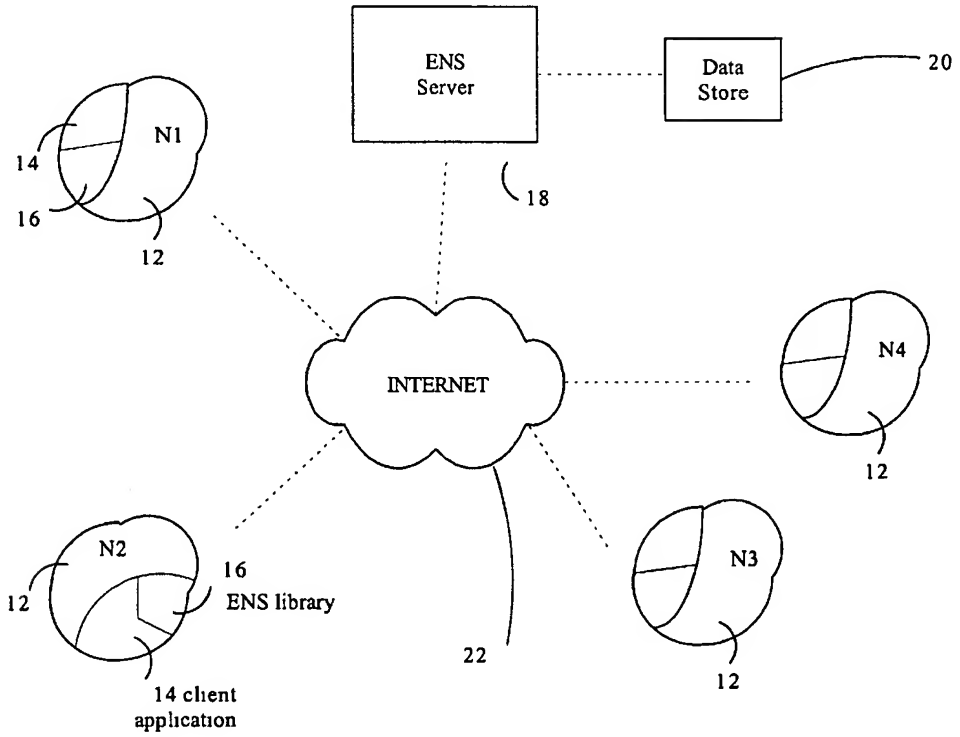
(51) Cl.Int.^{7}/Int.Cl.^{7} H04L 9/32, H04L 12/22

(71) Demandeur/Applicant:
ETUNNELS INC., US

(72) Inventeur/Inventor:
POIER, M. SKYE, CA

(74) Agent: ORANGE & CHARI

(54) Titre : SYSTEME ET METHODE DE FOURNITURE D'UN RESEAU PRIVE VIRTUEL
(54) Title: SYSTEM AND METHOD FOR PROVISIONING A VIRTUAL PRIVATE NETWORK



(57) Abrégé/Abstract:
A system and method to enable the secure transfer of information between nodal points in a workgroup over a public network by facilitating the creation of a virtual private network (VPN). The system comprises at least 3 nodal points and where at least one of these nodal points is an ENS server. The system is configured so as to centrally manage the system and allow for the update, modification, and implementation of information for each node in the workgroup from an ENS server. The system further comprises a database linked to the ENS server and a client application having an ENS library located at each node. The client application communicates with the ENS library through an API, and where the library transmits data to and receives data from the ENS server.

Abstract:

A system and method to enable the secure transfer of information between nodal points in a workgroup over a public network by facilitating the creation of a virtual private network (VPN). The system comprises at least 3 nodal points and where at least one of these nodal points is an ENS server. The system is configured so as to centrally manage the system and allow for the update, modification, and implementation of information for each node in the workgroup from an ENS server. The system further comprises a database linked to the ENS server and a client application having an ENS library located at each node. The client application communicates with the ENS library through an API, and where the library transmits data to and receives data from the ENS server.

FIELD OF THE INVENTION:

The invention relates to a system and method of providing secure communications over an open network, and more specifically to the application of a virtual private network which runs on a diverse set of operating systems and hardware platforms and facilitates ease of use.

BACKGROUND:

Workgroup computing involves, by definition, the exchange of data between the nodes of the workgroup, a node being a computer connected to a network which can be identified with an individual, a set of resources (files, services, devices, etc) or a gateway. Often the tasks of a workgroup are of a sensitive nature, containing, for instance, confidential information on finances, business development plans or private email. The Internet (and its native IP protocol) has become ubiquitous as a means of connecting nodes in a workgroup computing environment as it is cheaper and more effective than traditional private networks, especially when the nodes are geographically disparate. However, with the adoption of a public networking infrastructure comes the risk that an unauthorised 3rd party with access to the physical data route between two nodes may intercept and reconstruct data transferred between them. To prevent interception, a mechanism is required to modify the transmission of data such that only the intended receiver may access it and the receiver can be guaranteed of the data origin. A virtual private network is a system for securing communications between computers over an open network such as the Internet. By securing communications between the nodes it is as if the computers were linked together on a private local area network (LAN). An Internet-based virtual private network (VPN) is virtual because although the Internet is freely accessible to the public, the Internet appears to the organization to be a dedicated private network. In order to accomplish this, the data traffic for the organization may be encrypted at the sender's end and then decrypted at the receiver's end so that other users of the public network can intercept the data traffic, but cannot read it due to the encryption.

A VPN can replace an existing private data network, supplement a private data network by helping relieve the load on the private data network, handle new software applications without

disturbing the existing private data network or permit new locations to be easily added to the network. A typical VPN connects one or more private networks together through the Internet in which the network on each side of the Internet has a gateway and a leased line connecting the network to the Internet. In these typical VPNs, the same protocol for each private network is used which makes it easier to communicate data between the two networks. To create the VPN, a secure communications path between the two gateways is formed so that the two private networks may communicate with each other.

In the case of Internet communications, the most commonly used software routines for the transport which involves addressing, routing, and data formatting is the Transmission Control Protocol (TCP/IP). The TCP/IP suite consists of four layers: an application layer, a transport layer, a network layer, and a physical layer. Each of these layers interacts only with the layer immediately above and below, and serves a specific set of functions. Security measures may be implemented at any of the layers. however, VPN's are generally built at the network level which is responsible for the routing of data packets and the addressing mechanism.

In order to establish secure communication between any two nodes on a virtual private network, each node must obtain by some means information ("configuration") including but not limited to:

- The identity and state of the remote nodes within the VPNs the node is a member of
- The location of the remote nodes on the logical network (IP address)
- The relationships between nodes (VPN membership)
- The keys to be used for encrypting and decrypting data transferred between nodes (one per node pair)
- Verification of the identity of the remote nodes

It is this securing of communication between nodes in a workgroup that the present invention is designed to facilitate.

Traditional VPN solutions are comprised of a number of tunnel termination devices which provide a central "hub" for VPN communication. Software is then deployed to nodes that wish to participate in a VPN, and the software is configured manually with the address of the tunnel termination device(s). The software is then executed in order to participate in the VPN. However, there are several disadvantages with respect to this technology. Each of the nodes may only be a member of one VPN at a time in the majority of implementations. This limits the ultimate efficiency of the user at each node. In addition, the network does not have the ability to automatically configure nodes for VPN participation.

The Microsoft® Active Directory is a directory service designed to extend Microsoft®'s Windows NT "domain" architecture. It allows organizations to share and manage information about network resources and users. In addition, Active Directory acts as the central authority for network security, permitting the Windows operating system to verify a user's identity and control access to Microsoft® network resources. It also acts as an integration point for bringing systems together and consolidating management tasks. As part of the authentication process for "logging into" a domain, Active Directory allows a system administrator to specify security relationships to be automatically established by the authenticating node to other nodes within the NT domain. Using this technology has several drawbacks in that each of the nodes on the network must be running Win2000 Server or Professional Edition. Further, each of the nodes participating in the VPN must have accounts and be logged into the same NT domain which precludes automatic provisioning of cross-organizational or "extranet" VPNs. The system also poses greater security risks as it requires more TCP ports to be open through firewalls for automatic VPN provisioning.

The use of VPN's is well known in the computer world each using different mechanisms to provide a means of secure data transmission. United States Patent No. US6061796 entitled Multi-Access Virtual Private Network describes system and method for allowing private communication over an open network. This system however, specifies what mechanism protocol level the Agent (VPN provisioning application) uses to intercept incoming and outgoing data from a node and is not designed to work with IP networks. In addition, it would be difficult to scale this particular system for large-scale use. In United States Patent No. US5884035 and

6026430 data transmission is only through the domain hierarchy and not on a data to client application basis. In the VPN system described in United States Patent No. 6055575 it notes that the "host computer establishes a secure communications path, referred to as a tunnel, through the public network with the remote client". This has firewall implications in that a remote node can rarely accept incoming connections.

It is an object of the present invention to obviate and mitigate at least some the aforementioned disadvantages of the prior art.

SUMMARY OF THE INVENTION:

It is accordingly a principal objective of the present invention to establish a network based client-server system which creates security relationships that facilitate the instant and dynamic provisioning of VPNs.

This invention provides a system for facilitating the secure communication between nodes in a workgroup by the creation of a "n"-tiered virtual private network. Each node has the ability to transmit and receive data over a public network such as the internet. The system comprises at least 3 nodes and where at least one of these nodes is an ENS server, a data store linked to the ENS server, and a client application including: an ENS library located on each node and a mechanism to transmit data within the workgroup. The data store further includes information pertaining to the configuration of virtual private networks (VPN), VPN relationships (eg. client computers membership to VPN's), settings & options (eg. encryption level, IP compression), authentication information, objects & attributes (eg. status - online/offline, full user name, user IP), and keys. The system further includes a means to intercept both incoming and outgoing data from a node so as to create a secure bridge between an open network and a node by encrypting and decrypting data. In addition, the system haS a means for verification of node credentials against authentication servers. The bridge enables data to be securely shared to a private group without moving the data from a nodes local harddrive. Each of the nodes has the ability to act as both a client and a server.

In a preferred embodiment, on start up of a node within the workgroup, the client application instructs the ENS library via an API to form a connection with the ENS server. The API transmits authentication credentials to the ENS server, the ENS server authenticates the validity of the credentials, and a connection is established. Following the creation of a secure connection between the ENS server and a node, the API synchronizes with the ENS server and retrieves a packet of information from the data store that is transmitted to the ENS library. This information packet includes a list of VPN's of which this particular node is a member and their respective attributes, a listing of other nodes which are members of the same VPN as the client computer, the current status of each node in each respective VPN, and other related details. Once a node, say node 1, is logged onto the ENS server and, in turn, a desired VPN, the ENS library of node 1 sits in the loop between the ENS library of the client computer and the ENS server, the central management of the system enables the ENS server to be informed of any changes to a VPN eg. a node logging off, and is informed of these changes instantaneously. The ENS server then relays this information to the ENS library of each client computer within a VPN and the library, in turn, sends a message to the client computer reporting the change to the VPN.

This system is global by the nature of the ENS server such that it facilitates the central management of any VPN. The ENS server facilitates the ability to make changes to a VPN without having to effect changes manually in each node of a virtual private network. A change made to the data store linked to the ENS server is transmitted in real time to all client computers effected by the change. For example, to change the password of a VPN for each node in that network, requires making that change to the data store and, in turn, that change is transmitted to each node on the virtual private network. While changing a password is a relatively simple task, the ability to effect more detailed changes to a VPN requires updating only a single point in a VPN and then transmitting that data to the remaining nodes in the workgroup via the secure connection. In use, the network has the ability to automatically and securely provision security associations between nodes.

In its simplest configuration the ENS system employs security measures at the transport layer of the TCP/IP however, any layer or combination of layers of a transport protocol may be used by the interception means in order to transfer data. Generally, the client application sees the

destination IP address in the header of the data packet, if that address is a node within the VPN, a key is applied to encrypt the data which is then sent to the appropriate node. The process works the same way for decrypting. The key for encrypting and decrypting messages between two nodes is the same. This is called END-to-END security. When a node initially logs onto to a VPN and receives the information packet from the ENS server, that information contains keys for a node within a VPN to encrypt and decrypt messages sent within that VPN network. The method of encryption and decryption employed is beyond the scope of this invention.

The control of the VPN created using the ENS server may be in house in the sense that at a particular company subscribing to this service, the IP manager would administer and maintain the VPN and have rights to modify information on the ENS server and data store as it pertains to their VPN. Relationships between nodes in a VPN are pair based. Any and all traffic between two nodes on a VPN is encrypted and decrypted regardless of the type of information being sent. The ENS server itself does not participate in the data transfer.

Embodiment of the invention will now be described by way of example.

Description of the Drawings:

Figure 1: Is a schematic diagram of an overview of d computer system;

Figure 2: Is a functional block diagram detailing the method for establishing secure communication between nodes, in the computer system of figure 1;

Figure 3: Is a schematic of the computer system incorporating a plurality of types of nodes;

Figure 4: Is a schematic diagram of the computer system detailing use of a gateway node in co-operation with the plurality of other nodes; and

Figure 5: Is a functional block diagram detailing the communication hierarchy of the interaction of the API with the computer system of the present invention.

Prior to the detailed description the following list of terms will be used to herein, these terms are said to have the following meaning:

Client Application - the software that acts as a slave to a server and is present on each node within a work group;

VPN - a virtual private network that is constructed over a public network to connect nodes within a work group such that:

a) data transferred between those nodes is secure and cannot be intercepted on route; and

b) it contains mechanisms to ensure that only authorized users may access the network..

A VPN exists for each work group (which subscribes to the eTunnels ENS service)containing nodes in that work group.

Node - a computer connected to a network which maybe identified with an individual, a set of resources, or gateway;

Work Group - a group of two or more individual nodes working collaboratively on a group of tasks;

Library - this is the software portion of the client application, it provides a library of functions and includes configuration data received from the server during the synchronisation phase to be described hereafter;

API - application program interface, designed for communicating between software applications; and

Gateway - a special node that provides secure communication to a specific network of nodes located behind the gateway.

DETAILED DESCRIPTION OF THE EMBODIMENT

A computer system for establishing a secure connection for the transfer of data between nodes in a work group over a public network is illustrated in figures 1 through 5 and is generally designated by reference numeral 10.

As shown in Figure 1, a computer system 10 comprise a plurality of nodes 12, an server 18, and a data store 20 whose contents may be updated a changed periodically by external intervention.

Each of the nodes 12 has a client application 14 and an Library 16. The system 10 enables the establishment of a secure path for communication between nodes 12 over a public network such as the internet 22. The server 18 collects and distributes data collected by the library 16 at each node 12, so as to maintain state information for each node 12. The server 18 tracks changes made to the data store 20 and subsequent by updates each of the libraries 16. The library 16 communicates with the client application 14, and vice versa, through an application program interface (API), where the API is responsible for transmitting to and receiving from information client application 14, and related library 16, and the server 18. The server 18 also serves to generate specific node cues based on those events, such as the availability of upgrades for client application. The data store 20 is linked to the server 18, and is essentially managed so as to enable the automatic provisioning of security relationships with nodes 12 in a network. A network having secure communication between these nodes 12 is typically known as and from herein referred to "a virtual private network" (VPN). The centrally managed data store 20 allows for arbitrary additions, modifications, and alterations to that data store 20 and, inturn, deploys that information through the server 18, to nodes 12 located within a virtual private network.

The method of establishing secure communication between nodes in a work group is detailed in Figure 2. On startup of a node within a work group, the client application 14 instructs the library 16, through the API to form a connection with the server 18. Once the instructions have been received, as indicated at 102, the API forms a socket connection, generally using secure socket links (SSL/3DES socket security). Once a connection, 104, is formed between the server and a node, the authentication phase, 106, begins. The client application transmits credentials through the library 16 to the server. The server then authenticates the validity of these credentials and returns data stating the success 108 or failure 109 of the logon to the server. If the credentials are found to be invalid the process fails and ends.

Once the node is logged onto the server 18 the synchronized phase 110 begins. The server 18 delivers a packet of information to the library 16 via the API so as to establish a virtual private network. The data packet includes, but is not limited to, a list of virtual private networks to which that node is a member, their related attributes, the state of other client computers located within a virtual private network of which the client computer is a member, and their related

details such as IP address, encryption/decryption keys etc. Once this transfer of information 112 has occurred, the server 18 and node 12 are successfully linked as indicated at 114, and the transfer of data over secure line of communication is enabled.

Thereafter any change to the data store 20 that affect a work group of which the node is a member will be forwarded from the server 18 to that node. The server is able to determine the relevant nodes 12 from the contents of the data product received during the information transfer phase 112.

Figure 3 illustrates a plurality of nodes 12A through 12E, where at nodes 12C through 12E there are a plurality of client computers. The computer system 10 detailed in Figure 3 is a multi tiered client/server system in which every node 12 acts as both a client and server. The server 18 operates over an existing network connection to the public network connection 22 that each node 12 possesses. The computer system10 allows arbitrary grouping of nodes 12 on the Internet 22 into VPNs across, for instance, network, organisational and geographical boundaries.

The computer system 10 enables an extra net connection for example between two offices of a company 12D and 12E, each of which has its own intranet, to be included a work group. In this situation a corporation typically will have at least one localized server 17B, 19B, which will act as server for that intranet. Each client computer within that corporation will be connected to that localized server. The localized server 17B, 19B exists within a hierarchy within the computer system such that if a node/client computer within the corporation queries the localized server, and that server does not contain the information queried for, that server climbs the hierarchy chain to a higher up server and queries for the information. This process continues until the information is returned to the localized server where it can be distributed to the appropriate client computers within that network. Alternatively, a node within the corporate network is capable of communicating with, for example a traveling user 12B located outside the office.

When each node 12A through 12E logs onto the server 18, it receives a list of keys, such that each node in the network exists in a parallel relationship with another node. Each pair of nodes is typically setup with a set of keys and a unique identity such that they may transmit secure

messages that have been encrypted and decrypted using this set of pair based keys. It will be appreciated that when transmitting data between two nodes logged on to a virtual private network, that data is not transmitted through the server 18. The server 18 is used for the initial provisioning of the virtual private network and to transfer information through the API to the library 16 of each node with configuration information for the provisioning of that virtual private network.

Figure 4 again shows computer system 10 and in this embodiment involves the use of a gateway 24 that includes a library portion containing attributes of the servers connected to the gateway 24. The gateway 24 is considered a node by other users and has a key pair associating it with each of the other nodes.During the logon process detailed in Figure 2 the server 18 will detect the presence of the gateway 24 and during the synchronization phase the data store 20 will provide information to the library portion 16 of the gateway 24 as to the range of IP addresses that are assigned to nodes behind the gateway. The gateway 24 has a set of rules called security associations that are designed to control access to the VPN such that the gateway protects a plurality of nodes. When a node in front of the gateway, such as 12A wishes to communicate with a node behind the gateway such as 12G, the node 12A selects the key pair associated with the gateway 24 to provide encryption and decryption of the data. The decryption then occurs at the gateway as opposed to at the node. When a user who is typically part of the plurality of nodes located behind the gateway, such as a company network, is working from home, the IP address of the home computer 12A is not in the range of IP addresses specified by the gateway. When an IP address falls outside the range of addresses known to the gateway access maybe denied to the company network. In such a situation, a virtual IP address is typically assigned to the home user 12A. When a VIP is assigned to the node of the home user 12A, data being send from node 12A to the company network 12G located behind the gateway will route to this data through a virtual interface. In the case where a node is a intranet as in Figure 3 node 12C, the server 18 will have a plurality of rules stating which client computers may access data on the servers, this set of rules known as an access control list (ACL).

Figure 5 shows the functioning of an application program interface (API). The API is designed for transparent communication between nodes/client computers and various internet services.

The API 202 is basically a configuration manager which generally operates through a remote procedure call protocol in order to communicate demands between an application program and a protocol stack. An API may also facilitate communication with other programs not shown here. Prior to the transport of data packets between the protocol stack and the hardware driver a shim intercepts a data pack in order to authenticate the parties to the communication by establishing a secure channel between the server 18 and the nodes communication. The shim communicates with the library 16 through a plurality of security authorities in order encrypt and decrypt the data package to ensure the validity of the credentials.

On securing a communications path over a public network between two nodes in a computer work group, a typical encryption technique used to transfer data between these nodes includes: generating a data packet to be transmitted over the secured communications path where the data packet includes routing information, encrypting that data packet using an encryption technique known to one skilled in the art, encapsulating the encrypted data packet into a secondary data packet compatible with public network protocols, transmitting the encapsulated data packet over the public network, the data packet arriving at the receiving node, and that receiving node unpacking the encrypted data packet using a set of authentication keys, stripping the second data packet from the original data packet, and decrypting that data packet received from the originating node.

In the preferred embodiment, the system of the present invention is a TCP based n-tier client-server system in which each node is capable of acting as both a client and a server over a single connection. The Distributed Object Architecture (DOA) generally used is a 3rd party implementation of the CORBA 2.3 specification using ,for the most part, SSL encrypted BIIOP. In CORBA, interfaces to remote objects are described in a platform-neutral interface definition language. This choice of protocols satisfies the requirement that the protocol operate through most firewalls, since encrypted data cannot be inspected in transit. In addition, the system will use port 443 (fallback to port 7030 if unavailable), such that the ENS session will appear to be a HTTPS session to firewalls. The use of BIIOP allows both peers to act as a client and server over a single connection which the Agent initiates to an ENS server, which eliminates more potential firewall problems as incoming socket connections are rarely allowed. The use of the CORBA 2.3

specification, and in particular the new portable object architecture (POA) layer, ensures independence from a particular CORBA implementation such that future implementations may be changed without serious impact on the ENS codebase.

The SSL encryption at the BIIOP layer typically utilizes server-side certificates in order to establish upstream trust relationships between ENS peers. A primary data object of the computer system described above is the platform independent servers and related databases, from herein referring to as ENSDataModel class, which contains the following objects and their related methods and attributes:

- nodes (user, server, gateway)
- relationships (VPN membership)

Each node of the computer system may instantiates the ENSDataModel class, which abstracts the use of CORBA to communicate with remote instances of the ENSDataModel class.

The ENSDataModel object provides methods for iterating over and retrieving nodes. In addition, the ENSDataModel generally provides two mechanisms for receiving notification of changes to the ENSDataModel. First, methods for "subscribing" to a particular node are provided such that if the target object changes within the filter rules specified, a registered callback is invoked. In the second method, a ENSDataModel-wide change callback may be registered, again with filter rules to narrow the notifications to those of interest. This mechanism not only serves the client application but is the mechanism by which the ENSDataModel propogates notification of changes to objects through the hierarchy of nodes.
Another object class in the computer system is denoted is ENSAuth. This object allows the client application to submit its authentication credentials and receive a token that is supplied to the ENSDataModel in order to gain access to the data store. Authentication credentials may also allow a node to act as the authenticator for a particular node or network. In this case, the request will be routed to this node for approval by modifying the appropriate attribute for the object that this node wished to be responsible for authenticating. Authentication may take one of several forms. In its most basic form, credentials are a double consisting of <ID, password>. Some credentials, such as those used for servers or gateways, may be IP restricted. In this case, the credentials are a triple

consisting of <ID, password, IP address>. In addition to the above scenarios, the use of certificates may be employed. If the attributes of a particular VPN specify the use of certificates, then the users's "self" object will indicate that a public/private key pair require generation and the ENSDataModel will deny any requests for data. The public key may then be submitted to the PKI database of the upstream ENS server (through the appropriate ENSDataModel method). The node is then forced to re-authenticate using a signed signature, and checked against the stored public key. The verification of node credentials may be against internal or external authentication servers.

The computer system 10 may be run on a diverse set of operating systems and hardware platforms such as open BXD, UNIX, Windows NT, Windows 95/98, Linnux, and Solaris.

CLAIMS:

1. A method for establishing a system for the secure communications between nodes in a workgroup over a public network by facilitating the creation of a virtual private network, the method comprising the steps of:

said method comprising the steps of:

a) a client application instructing a library of one of said nodes to form a connection with a server;
b) said server authenticating the validity of a credential of one of said nodes and returning data allowing said node to logon to said server;
c) synchronization of one of said nodes with said server such that said library of receives a list of VPN's to which said node is a member, their related attributes, status of additional nodes located within the VPN's of which said node is a member, and the related attributes of said additional nodes; and

wherein the data store of said server is centrally managed so as to enable the automatic provisioning of security relationships with nodes in a VPN.

2. The method for establishing the computer system of claim 1, wherein said server enables the automatic and secure exchange of configuration information between said nodes within said workgroup and restricts configuration exchanges based on trust relationship established by said node credentials.

3. The method for establishing the computer system of claim 1, wherein said library of each of said nodes remains in a loop with said server so as to relay any changes within a VPN to each node within that VPN automatically.

4. The method for establishing the computer system of claim 2, wherein said server automatically sends changes within a VPN to each of said nodes in that VPN.

5. The method for establishing the computer system of claim 2, wherein said client application of one of said nodes automatically pull changes within a VPN from said server so as to update said library of said node.

6. A computer system for establishing secure communication between nodes in a workgroup over a public network by facilitating the creation of a virtual private network, the system comprises:

at least three of said nodes in said workgroup connected by said network;

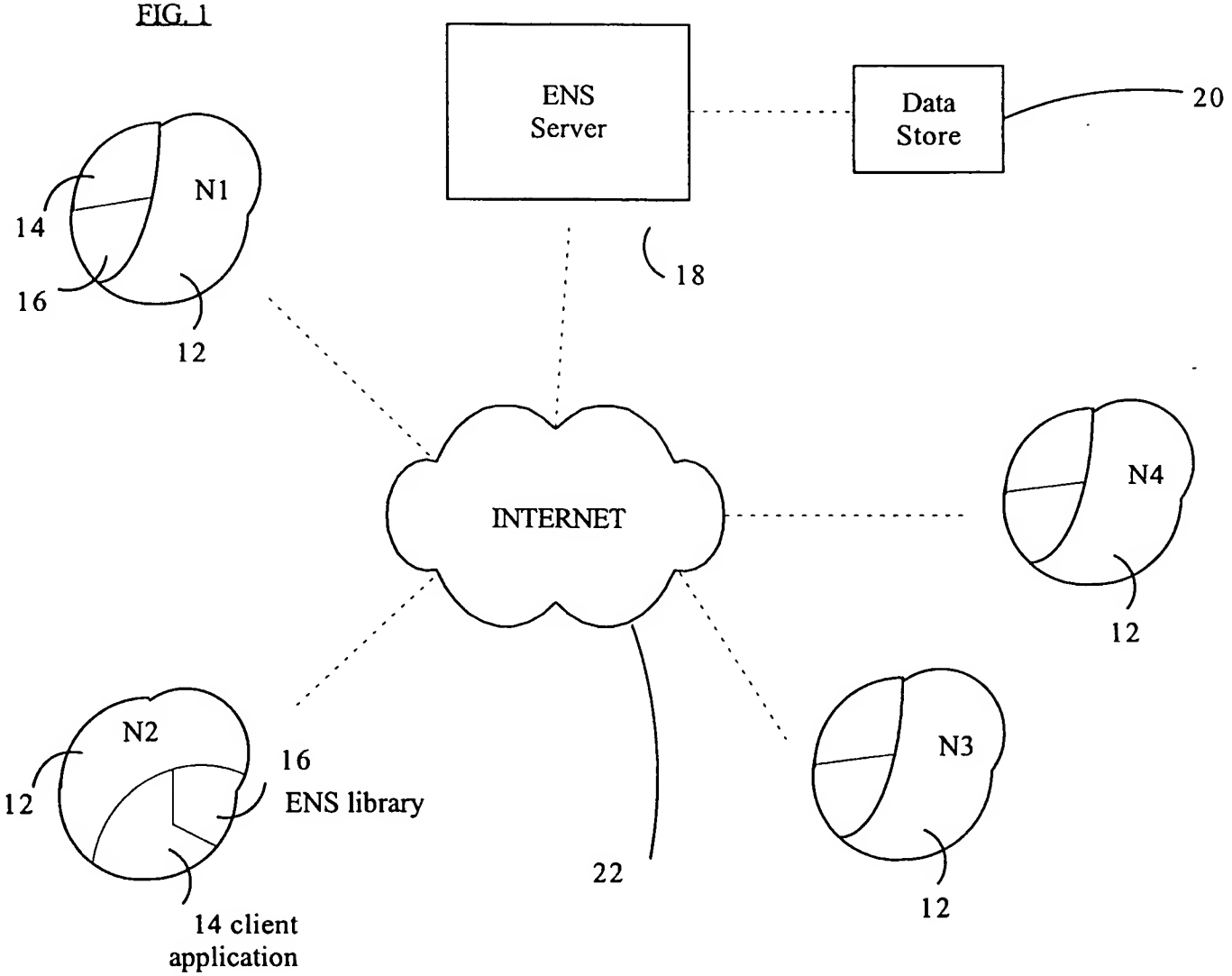
one of said nodes is a server, said server for monitoring the automatic distribution of data collected by each of said nodes, said server having a data store linked thereto; and

a client application software located at each of said nodes, said client application software including a library having functions related to the operation of said server, said client application capable of communicating with said library through an API;

wherein the data store of said server is centrally managed so as to enable the automatic provisioning of security relationships with nodes in a VPN.

7. The computer system of claim 6, wherein said server monitors arbitrary additions, modifications, and alterations to said data store and deploying that information through said server to said nodes within a VPN.

FIG. 1


ENS
Server
Data
Store
20
N1
14
16
12
18
INTERNET
N4
12
N2
12
16
ENS library
14 client
application
22
N3
12

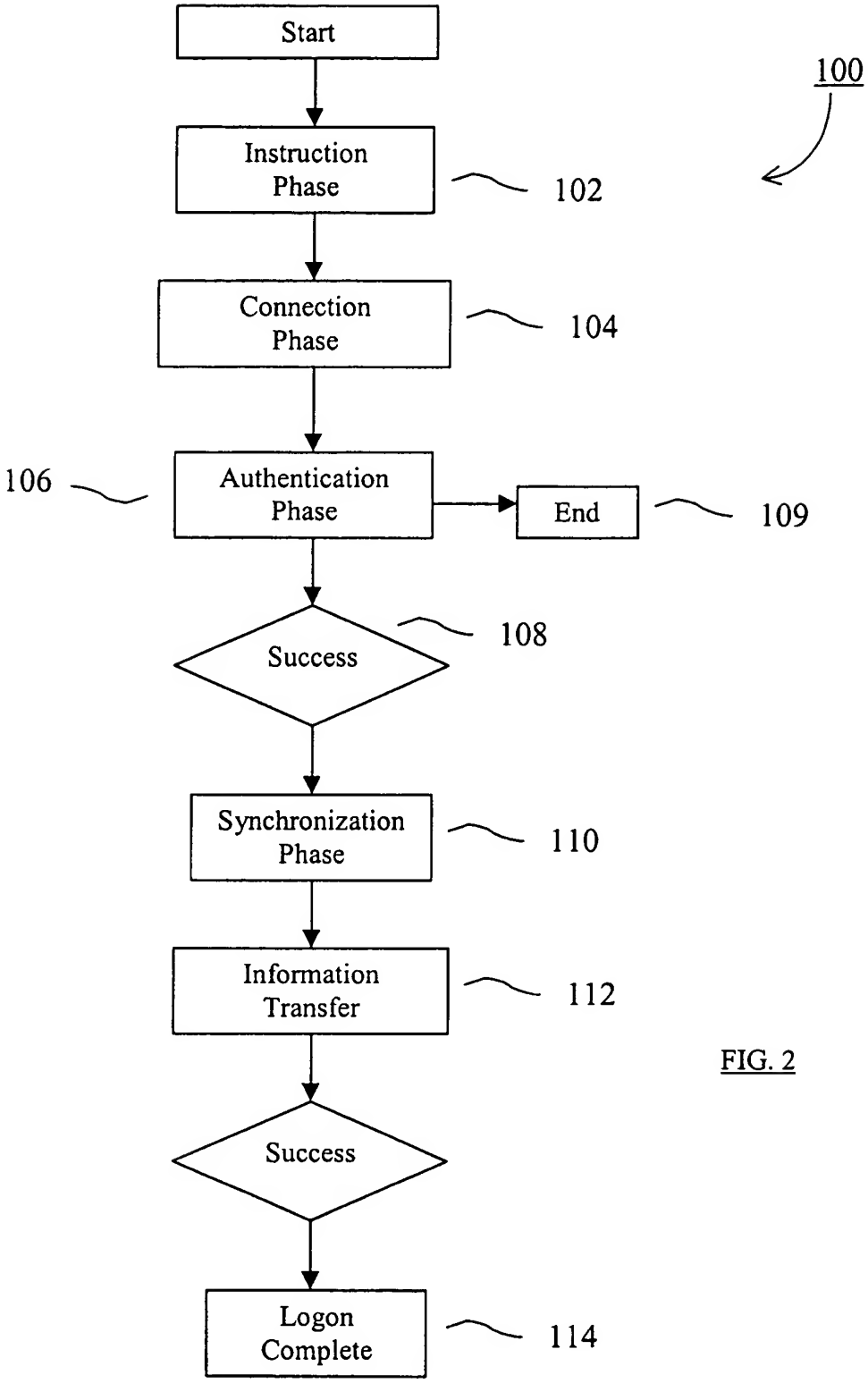

Start
100
Instruction Phase
102
Connection Phase
104
106
Authentication Phase
End
109
Success
108
Synchronization Phase
110
Information Transfer
112
Success
Logon Complete
114

FIG. 2

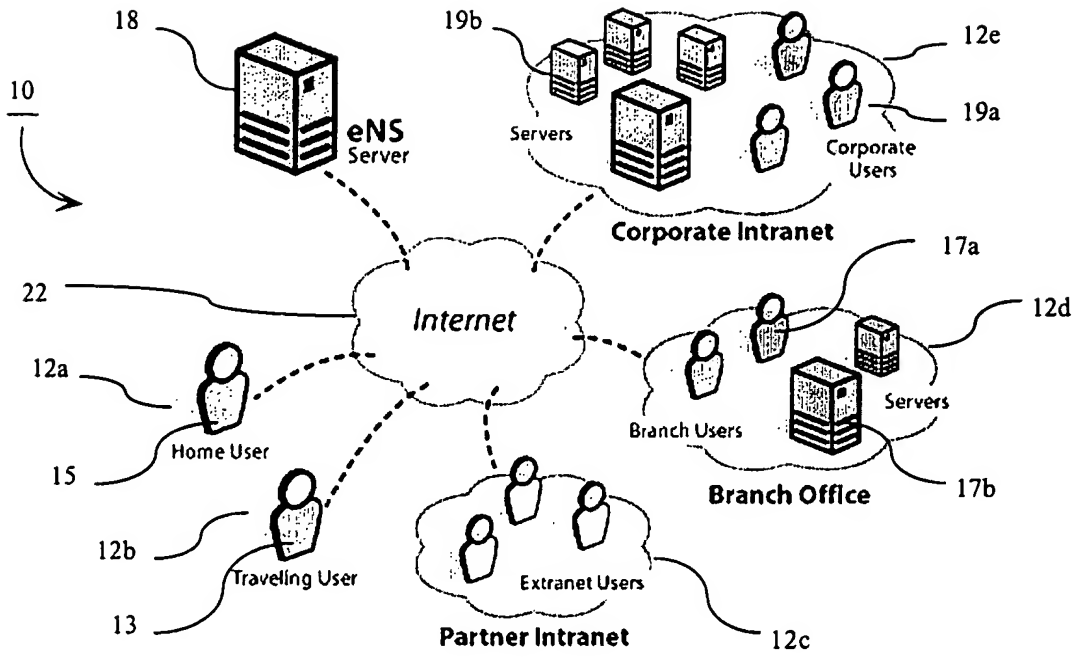

18
19b
12e
10
eNS
Server
Servers
19a
Corporate
Users
Corporate Intranet
17a
22
Internet
12d
12a
Servers
Branch Users
Home User
15
17b
Branch Office
12b
Traveling User
Extranet Users
13
Partner Intranet
12c

Fig. 3.

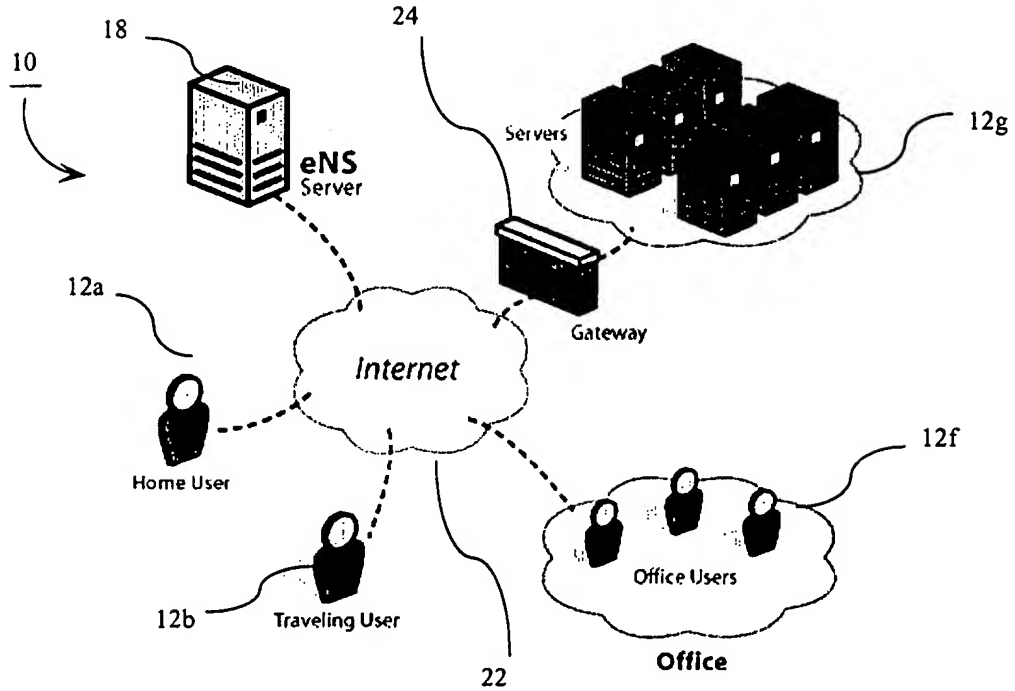

18
24
10
eNS
Server
Servers
12g
Gateway
12a
Internet
Home User
12f
Office Users
12b
Traveling User
22
Office

Fig. 4.

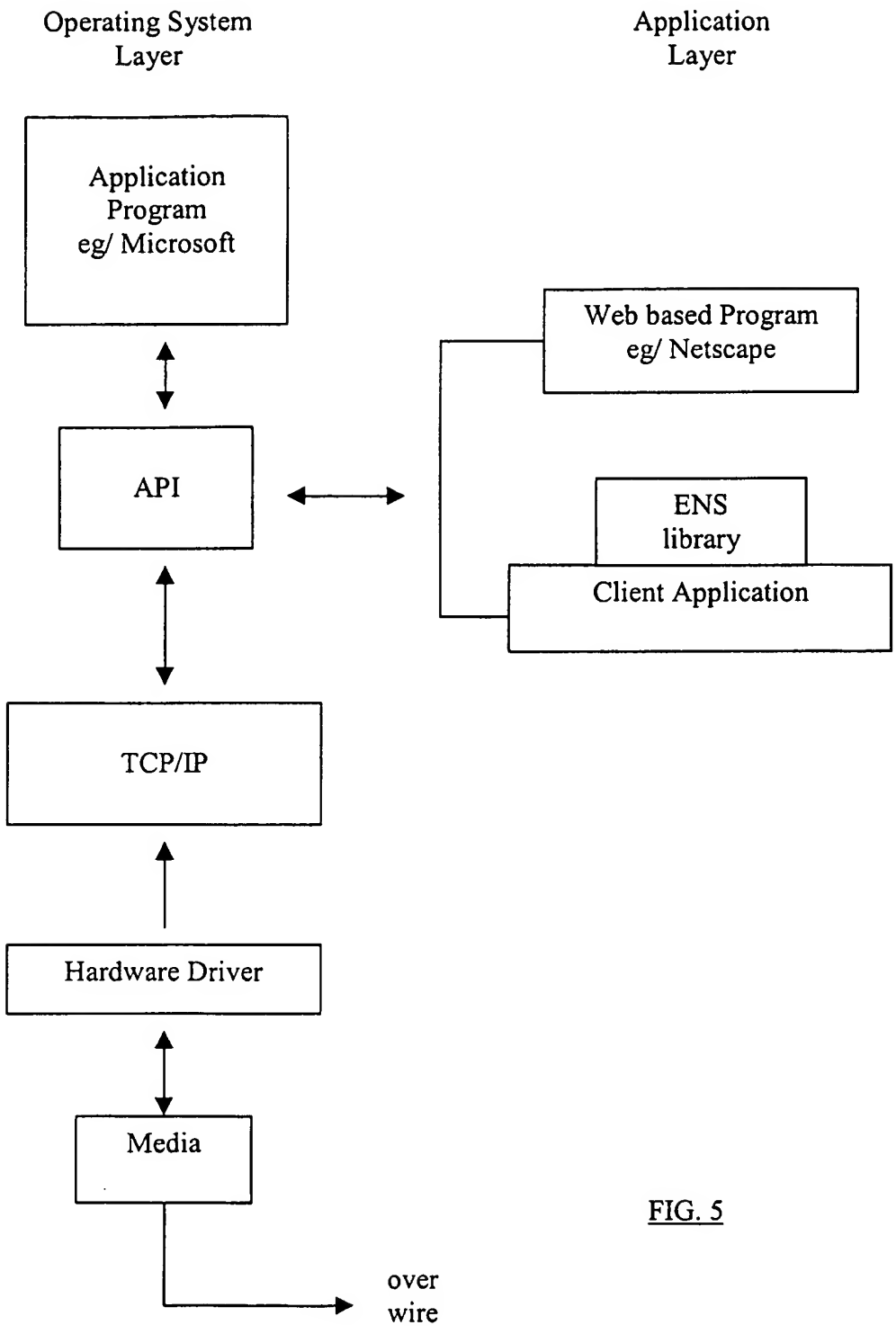

Operating System Layer
Application Layer
Application Program eg/ Microsoft
Web based Program eg/ Netscape
API
ENS library
Client Application
TCP/IP
Hardware Driver
Media
over wire

FIG. 5